AMENDMENTS TO THE CLAIMS

1. (Currently Amended) A method for dynamically changing an intrusion detection rule in a kernel level intrusion detection system, the method comprising the steps of:

a) generating a replica of ~~the~~ a current intrusion detection rule in a kernel area;

b) setting a first global variable when changing the replica of the current intrusion detection rule into a new intrusion detection rule, in response to a request from a user area for changing the intrusion detection rule;

b1) setting a second global variable and resetting the first global variable after the replica is changed ~~set of global variables after changing the replica~~ to indicate to a packet received after step b) that a change to the intrusion detection rule is in process and the packet is to use the new intrusion detection rule; and

c) resetting the second global variable when ~~changing~~ exchanging the current intrusion detection rule with the replica ~~by exchanging a value of a pointer representing the intrusion detection rule with a value of a pointer representing the new intrusion detection rule~~ and using the new intrusion detection rule on the packet.

2. (Previously Presented) The dynamic changing method as recited in claim 1, further comprising the step of d) generating a replica of the new intrusion detection rule.

3. (Currently Amended) The dynamic changing method as recited in claim 1, wherein in the step b) and the step c), a change state of the intrusion detection rule with a set of pre-assigned global ~~variable~~ variables is shown and the current intrusion detection rule is changed according to the set of pre-assigned global variables~~variable~~.

4. (Original) The dynamic changing method as recited in claim 3, wherein the kernel area transfers the request of changing the intrusion detection rule from the user area by using a system call.

5. (Previously Presented) The dynamic changing method as recited in claim 3, wherein the kernel area transfers the intrusion detection result to an application program of a

host, the intrusion detection rule being applied to the intrusion detection result, the intrusion detection result being transferred by setting the set of global variables inside the kernel and determining the transferring position inside the kernel.

6. (Currently Amended) A computer-readable medium storing program instruction for executing a method for dynamically changing an intrusion detection rule in a kernel level intrusion detection system, the method comprising the steps of:

a) generating a replica of ~~the~~ a current intrusion detection rule in a kernel area;

b) setting a first global variable when changing the replica of the current intrusion detection rule into a new intrusion detection rule, in response to a request from a user area for changing the intrusion detection rule;

b1) setting a second global variable and resetting the first global variable after the replica is changed~~set of global variables after changing the replica~~ to indicate ~~to~~ to a packet received after step b) that a change to the intrusion detection rule is in process and the packet is to use the new intrusion detection rule; and

c) resetting the second global variable when ~~changing~~ exchanging the current intrusion detection rule with the replica~~by exchanging a value of a pointer representing the intrusion detection rule with a value of a pointer representing the new intrusion detection rule~~ and using the new intrusion detection rule on the packet.

7. (Previously Presented) The computer-readable medium as recited in claim 6, further comprising the step of e) generating a replica of the new intrusion detection rule.

8. (Currently Amended) The computer-readable medium as recited in claim 6 or 7, wherein in the step b) and the step c), a change state of the intrusion detection rule with a set of pre-assigned global ~~variable~~ variables is shown and the current intrusion detection rule is changed according to the set of pre-assigned global ~~variable~~variables.

9 (Original) The computer-readable medium as recited in claim 8, wherein the kernel area transfers the request of changing the intrusion detection rule from the user area by using a system call.

10. (Previously Presented) The computer-readable medium as recited in claim 8, wherein the kernel area transfers the intrusion detection result to an application program of a host, the intrusion detection rule being applied to the intrusion detection result, the intrusion detection result being transferred by setting the set of global variables inside the kernel and determining the transferring position inside the kernel.